**PCT** WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : H04L 9/00 — A1

(11) International Publication Number: WO 98/42098

(43) International Publication Date: 24 September 1998 (24.09.98)

(21) International Application Number: PCT/US98/04658

(22) International Filing Date: 11 March 1998 (11.03.98)

(30) Priority Data:
08/818,132 14 March 1997 (14.03.97) US

(71) Applicant: CRYPTOWORKS, INC. [US/US]; 2084 Union Street, San Francisco, CA 94123 (US).

(72) Inventor: LeBOURGEOIS, John, H.; 193 San Carlos Way, Novato, CA 94945 (US).

(74) Agent: WOLFELD, Warren, S.; Fliesler, Dubb, Meyer and Lovejoy LLP, Suite 400, Four Embarcadero Center, San Francisco, CA 94111-4156 (US).

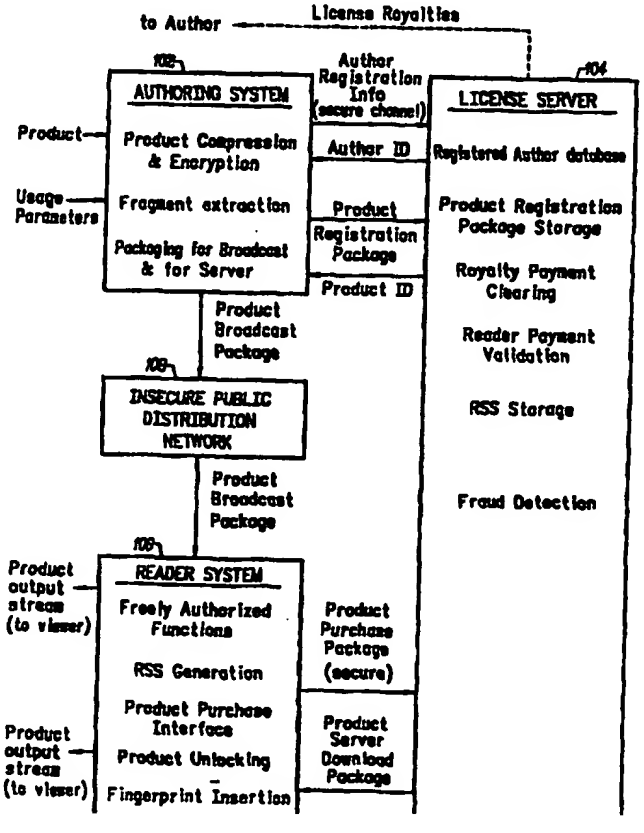
(81) Designated States: AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GE, GH, GM, GW, HU, ID, IL, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, UZ, VN, YU, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG).

**Published**
*With international search report.*

(54) Title: DIGITAL PRODUCT RIGHTS MANAGEMENT TECHNIQUE

(57) Abstract

A digital product is freely distributed through uncontrolled channels in encrypted form (108). Security fragment(s) of the encrypted product are withheld (102), and provided only upon communication with license server (104). The customer uses reader software (106) to purchase a license. Such software (106) examines components then present on the reader system to develop a reader system signature, which the license server (106) uses to encrypt the product decryption key and the security fragments before sending them to the reader system. When the customer wishes to use the product, a new reader system signature is generated and used to decrypt the product fragments.



encrypted form. The customer's software then uses a "secret key" to decrypt the product and use it. Secure direct modem connections may be used instead of Internet connections at various parts of the process in order to further ensure that no one else can use the encrypted product.

The CDI technique suffers from at least two problems. First, while the encryption of a data product during transmission may be very strong, it is still theoretically possible for an unauthorized third party to decipher it. Second, even if the product remains secure during transmission, once it is decrypted by the customer, CDI's strong encryption techniques no longer protect the product against further unauthorized distribution. The above-cited documents indicate that content as well as executables can be "locked" to a specific registered machine while still allowing for unrestricted distribution of the software in a crippled or time-dated mode, but the documents do not explain how this is to be accomplished.

In Schull U.S. Patent No. 5,509,070, incorporated by reference herein, Schull describes a technique for distributing digital products (specifically software) by selling a password to the user. The user uses the password to unlock advanced features of the product, but the password is usable only on one machine. Thus, the technique allows users to freely distribute software to other machines and other potential users, but does not allow other users to take advantage of advanced features of the software without obtaining a new password which is specific to the new machine. In the Schull method, the user generates a "passwordable-ID" either from the user's voice, by reading the serial

visited November 19, 1996, and all incorporated herein by reference). However, all of the technologies described in these references suffer from one or all of the problems mentioned above, as well as others.

Accordingly, there is a deeply felt need for a new technology which will control the distribution of digital products via the Internet and other uncontrolled distribution channels such that a fair return to the originator of the product can be ensured without unduly hampering wide and free distribution of sufficient information about the digital product to enable customers to decide whether to purchase a license.

## SUMMARY OF THE INVENTION

According to the invention, roughly described, a digital product is freely distributed through uncontrolled channels in encrypted form. Unencrypted preview material may also be provided in order to help the customer decide whether to purchase a license. In an aspect of the invention, one or more fragments of the encrypted product are withheld from uncontrolled distribution, and provided only upon communication with a license server. Unlike prior art mechanisms, which rely on practical limitations of computing power in order to prevent unauthorized product decryption, the technique of the present invention renders it literally impossible for an eavesdropper to recover the complete decrypted product without contacting the licensor. The product is not merely encrypted; to the extent of the security fragments, it is not even there.

In a second aspect of the invention, again roughly described, the customer purchases a license through the use of reader software which examines the

individual component signatures are then combined to form the overall reader system signature, for example by a weighted sum of the individual component signatures or by concatenating the individual component signatures together. If the reader system signature is determined on the basis of a weighted sum (or equivalently, a weighted average) of the individual component signatures, then the amount of permissible upgrade drift can be expressed as a percentage; that is, if the reader system signature generated upon usage of the digital product differs from the reader system signature generated at the time the product is purchased by no more than a predetermined percentage or fraction, then the usage is considered authorized. If the reader system signature is generated as a concatenation of the individual component signatures, then the number of components which differ at usage time relative to purchase time can be specified not to exceed a specific count.

In a situation where the reader system signature generated at the time of purchase is not stored on the reader system, it can instead be uploaded to a license server. If the reader system signature generated at usage time is found by the reader system not to properly decrypt either the product decryption key or the product itself, then in an embodiment, the reader system can automatically contact the license server for reauthorization. The reader system uploads the newly generated reader system signature, and the license server performs the upgrade drift test in comparison with the reader system signature that was stored on the license server at the time of purchase. If the license server determines that the newly generated reader system signature is within the permissible upgrade

Fig. 4 is a flow chart illustrating the flow of a product registration segment of Fig. 3.

Fig. 5 is a flow chart illustrating the general operation of the reader system of Fig. 1.

Fig. 6 is a flow chart of the product purchase preparation step of Fig. 5.

Fig. 7 is a flow chart illustrating one technique for generating the reader system signature.

Figs. 8 and 9 together constitute a flow chart of steps which takes place in the license server 104 in response to receipt of a product purchase package.

Fig. 10 is a flow chart of the step in Fig. 8 in which the license server processes the customer's payment information.

Figs. 11-13 together constitute a flow chart of the step in Fig. 5 in which the reader system plays the digital product.

Fig. 14 is a flow chart illustrating the license server's operations in response to receipt of a re-validation package.

Figs. 15 and 16 are alternative details of the step in Fig. 14 in which the license server determines whether the difference between the two RSS's exceeds a threshold.

## DETAILED DESCRIPTION

Fig. 1 is an overall symbolic diagram of a system according to the invention. The system has three primary components: an authoring system 102, a license server 104 and a reader system 106. In addition, the overall system is most useful when used with an uncontrolled distribution channel such as an insecure public distribution network 108 (e.g., the Internet). In general operation, the author or proprietor of one

customer can purchase. If the customer chooses to purchase one of the license options, the reader system 106 examines certain components of the reader system and, in dependence thereon, generates a reader system signature (RSS). The reader system assembles a product purchase package including the RSS and payment information, and uploads it to the license server 104. The license server 104 processes the payment information and, if successful, transmits a product server download package back to the reader system. The reader system uses the product server download package to unlock the functions of the digital product which are authorized under the license option that the customer has purchased, and allows the user to use the product accordingly. In addition, the reader system 106 performs fingerprint and/or watermark insertion as described hereinafter.

The license server 104 performs a number of functions, including maintaining a database of registered authors and storing all of the product registration packages. The license server 104 also stores reader system signatures from customers, performs customer payment validation, as well as certain fraud detection functions as described below. The license server 104 also performs the functions of royalty payment clearing. Specifically, license royalties received from (or on behalf of) customers are properly accounted for and transferred to the proper authors after deduction of a commission.

In Fig. 1, the authoring system 102, the license server 104 and the reader system 106 are each illustrated as a respective individual block. Depending on the embodiment, each block might contain no more than a single computer, or in different

102, the license server 104 or the reader system 106 is stored on the disk drive controlled by the disk drive controller 222, and brought into main memory 210 as needed for execution. The computer system of Fig. 2 communicates with the other systems of Fig. 1, and with the distribution network 108, if appropriate, via the network adapter 232.

Fig. 3 illustrates the overall system flow for the authoring system 102. The authoring system flow is generally divided into two segments: an author registration segment 302 followed by one or more product registration segments 304. In the author registration segment 302, the author (or other proprietor) of one or more digital products enters his or her identification information. Such information can include, for example, the author's name, address, Social Security or other tax ID number, password or other challenge information (for confirmation of identity during customer service calls), e-mail address and/or telephone number (step 306). In a step 308, the authoring system uses this information to create an author registration package which is transmitted, in step 310, to the license server 104. The license server 104 adds the author and the author's identification information to its registered author database, and in step 312, the authoring system 102 receives and stores and author ID from the license server 104. The communication between the authoring system 102 and the license server 104 in the author registration segment 302 should take place via digital certificate and one-time secure channel, or by secure, signed electronic mail.

Fig. 4 is a flow chart illustrating the flow of a product registration segment 304 (Fig. 3). In a step

performance degradation that compression/decompression often entails.

In a step 410, a product encryption key is generated. The key can be generated in any known manner; for example, by a pseudo-random number generator using a seed derived from the time period between two successive user key strokes. In step 412, the compressed digital product is encrypted using the encryption key developed in step 410. Again, any known key-based symmetric encryption algorithm can be used (as long as the correct complementary algorithm is used for decryption on the reader system 106). One such well-known encryption algorithm is DES, described in National Institutes of Standards and Technology, "Data Encryption Standard," FIPS Publication No. 46-1 (January 1988), incorporated by reference herein. Another is Triple DES (also known as DES-3), and yet another is RC-5. RC-5 is described in R. W. Baldwin and R. Rivest, "The RC5, RC5-CBC, RC5-CBC-Pad, and RC5-CTS Algorithms", INTERNET-DRAFT (March 1996), available from ftp://ftp.nordu.net/internet-drafts/draft-baldwin-rc5-00.txt, visited March 4, 1997, incorporated herein by reference.

The result of product encryption step 406 is referred to herein as an encrypted "version" of the digital product. As used herein, a "version" of a digital product is still considered to be the digital product, because it continues to include all the information of the digital product. A native "version" of a product also is nevertheless "the product". A digital product can exist in several versions, each of which is a reversibly processed version of the native version.

of the product even if such remaining portion can be decrypted.

In a step 418, the authoring system 102 generates a digest of the encrypted product less the security fragments. Again any digesting algorithm, such as SHA-1, can be used in this step. In step 420, the authoring system 102 creates a product registration package and transmits it to the license server in step 422. The license registration package can form part of a digital certificate in one embodiment. The license server stores the product registration package and returns a product ID to the authoring system 102 (step 424). The reason that the security fragments should be kept as small as possible is to minimize the storage capacity requirements of the license server. In a step 426, the authoring system 102 creates a product broadcast package for the digital product and makes it available (step 428) via any uncontrolled distribution network, such as the Internet.

The product registration package, which can also be digitally certified, includes the following items:

- author ID
- usage parameters (both the free usage parameters and the usage parameters at various purchasable options)
- the encrypted security fragments
- a product decryption key (complementary to the product encryption key of step 410)
- the digest of the encrypted product less security fragments
- digest of the full encrypted product

The product decryption key referred to above is whatever key is required to decrypt the results of the encryption process of step 412. For symmetric

As can be seen, whereas the great bulk of the product is transmitted via the uncontrolled distribution network 108, not only is it encrypted, but it is also incomplete so that even if it could be decrypted, for example by a powerful computer system, important fragments of the product simply are not there.

Fig. 5 is a flow chart illustrating the general operation of the reader system 106. When a user installs the reader software on the reader system 106, the reader software automatically generates an installation ID. The installation ID, which is stored on disk in the reader system 106, is a substantially unique identifier of the installation. The installation ID is stored in such a way that if the particular installation of the reader system software was to be copied to a different system, the installation ID would likely be copied as well.

In a step 502, the customer/user of the reader system 106 downloads the product broadcast package for a digital product that he or she wishes to examine. In a step 504, the customer performs one or more of the freely authorized functions, including a preview of the material in the digital product (if available). For example, if the digital product is a sound, the preview material might be a portion of that sound. If a digital product includes a movie, then the preview might be a portion of the movie, or a trailer. If the digital product is an image, then the preview material might include a thumb nail of the image. If the digital product is text, then the preview material might include an abstract.

In a step 506, the customer chooses to buy a license in accordance with one of the license options

by the customer occurs on the reader system 106 through the reader system software.

Fig. 6 is a flow chart of the step 510 (Fig. 5) in which the reader system 106 prepares the product purchase package. In a step 602, the reader system 106 generates a reader system signature (RSS) for the reading system 106. The manner in which the RSS is generated is described hereinafter. In a step 604, the reader system 106 retrieves the previously generated installation ID, and in a step 606, the reader system generates a digest of the encrypted product less the security fragments (as obtained from the product broadcast package). The digest performed in step 606 should be the same as that performed by the authoring system 102 in step 418 (Fig. 4).

In a step 608, the reader system 106 extracts the product ID from the product broadcast package and in a step 610, the reader system assembles the product purchase package.

The product purchase package includes the following items:

- product ID
- customer's installation ID
- customer's identification information (or privacy ID)
- customer's payment information
- customer's contact information (including information on where to send the product server download package)
- RSS of the reader system 106
- generated digest of the encrypted product less security fragments

predefined field size. In another embodiment, optionally after digesting, the individual component signatures are averaged or summed together to form the overall reader system signature. The individual component signatures can be weighted prior to combination, in order to reduce the impact on the reader system signature that would result from changes in components that are more frequently subject to upgrade or replacement.

In one embodiment, the reader system 106 generates the reader system signature in dependence upon component signatures from the following components, to the extent present in the system. Except as indicated below, most of the component signatures set forth in this list are readable either from the CMOS or from a device manager driver. This is only an illustrative list; other embodiments can refer to other components not on this list.

Hard Disk Drive

- drive ID
- numbers of cylinders, sectors and heads
- drive defective sector map (obtained from sector 0)
- drive name
- drive manufacturer

Floppy Disk Controller

- I/O addresses and settings
- interrupt assignments
- manufacturer name

against the installation IDs that have been stored previously on the license server for other product purchases. If a large number of purchases have been made using product purchase packages specifying the same installation ID, then it is likely that someone has altered an installation of the reader system software and is passing it around to different customers who are using it to purchase licenses. The same is true if the same license has been purchased several times from the same installation ID, or if several significantly varying reader system signatures have been stored in the license server's database in conjunction with the same installation ID. A number of other fraud detection mechanisms can also be employed. In any event, an investigation is warranted if step 810 suggests that an altered version of the reader system software might be being distributed.

The flow chart of Fig. 8 continues after step 810 with step 902 in Fig. 9, as indicated by the symbol "9" in both Figs. 8 and 9.

In Fig. 9, in step 902, the license server 104 further encrypts the already once-encrypted security fragments (from the product registration package) using the customer's RSS as a key. The key used in step 902 need not be the RSS exactly; it can be some other number which depends on the RSS. For example, it can be a digest reduction of the RSS from the customer's product purchase package. In any event, step 902 results in "double-encrypted" security fragments from the digital product.

In step 902, the product decryption key from the product registration package is also encrypted using the customer's RSS (or a number derived therefrom) as a key. Note that in a different embodiment, either

error to step 806 (Fig. 8) (Step 1006). If an approval code was received, then in step 1008, the license server 104 credits the author's account with the amount of the approved purchase price less a commission. In step 1010, the license server 104 returns successfully to the step 806 (Fig. 8).

Returning to Fig. 5, as previously mentioned, each time the customer desires to use the digital product, he or she does so using the reader system software on the reader system 106. Fig. 11 is a flow chart of the step 518 in which the reader system plays the digital product. (The terms "play", "view" and "use" are used interchangeably herein as regards a digital product.) Referring to Fig. 11, in a step 1102, the reader system 106 regenerates the RSS for the reader system. This step takes place using the same algorithm that was used in step 602 (Fig. 6) when the RSS was generated for preparation of the product purchase package. In a step 804, the reader system 106 decrypts the double-encrypted security fragments using the new RSS as a key. As mentioned with respect to step 904 (Fig. 9), the key used in step 1104 need not be the RSS identically; another number which depends on the RSS can be used instead. However, whatever algorithm is used to derive the key from the RSS in step 1104 should be the same as that used in step 904.

In step 1106, the reader system 106 merges the encrypted security fragments into the encrypted product less the encrypted security fragments, thereby assembling a complete, but still encrypted, version of the digital product. In step 1108, the full encrypted digital product is digested using the same algorithm as was used originally by the authoring system 102 in step 414 (Fig. 4). In step 1110, the reader system 106

decompressed using an algorithm complementary to that used by the authoring system in step 408 (Fig. 4). The resulting decompressed digital product is transmitted in step 1210 to an appropriate viewer.

It will be appreciated that once the digital product is transmitted in step 1210 to a viewer, which may be any standard viewer appropriate to the content of the digital product, the output stream is no longer secured by the mechanisms built into the overall system as described herein. Accordingly, a step 1208 is optionally inserted between steps 1206 and 1210 of Fig. 12. In an embodiment which includes step 1208, a fingerprint and/or a watermark is (are) inserted into the digital output stream prior to or while it is being provided to the viewer. Watermarking is a technique using a visible identifier that will let the user know that he or she has been associated with this particular instance of the content. It acts primarily as a deterrent. Fingerprinting embeds and hides codes into the output stream itself that are retrievable only by the author or by the licensing authority. Such codes uniquely associate the particular copy of the digital product with the individual who purchased it. Fingerprinting is used primarily for criminal prosecution and court proceedings.

If fingerprinting is used, preferably the fingerprint is inserted in a manner which does not affect the resulting viewing experience. For example, if the output stream includes CD audio, then the fingerprint can be spread over a large number of the audio samples, either substituting for the low-order bit or modifying the low-order bit in an exclusive OR manner in each sample. Alternatively, to avoid differential cryptanalysis, the data stream can be

package to the license server 104 at the URL identified in the product broadcast package. The license server's operations in response to receipt of a re-validation package are set forth in Fig. 14.

Referring to Fig. 14, in a step 1402, it is first determined whether the RSS in the re-validation package was based on a component in the reader system 106 having external assurances of substantial uniqueness. If so, then re-validation is considered unsuccessful (step 1404) and this result is returned to the reader system 106. If the RSS in the re-validation package was not based on a component having external assurances of substantial uniqueness, then in step 1406, the license server 104 compares the new RSS from the re-validation package to the RSS previously stored accessibly to the server for the same reader system 106 (as identified by the installation ID specified in the re-validation package). If the difference between the two RSS's exceeds the threshold that was specified by the author in the usage parameters stored on the server 104 for the product ID specified in the re-validation package (step 1408), then, again, re-validation is unsuccessful and such a result is returned to the reader system 106 (step 1404). In different embodiments, the threshold can be specified as a percentage of one or the other RSS, or as a number of component signatures which differ between the two RSS's, or by a number of other different specifications.

If the difference between the two RSS's does not exceed the designated threshold (step 1408), then the re-validation is considered successful. The license server 104 prepares a new product server download package using the same algorithms as set forth above

104 counts the number of components of the RSS in the re-validation package, which differ from the corresponding components of the RSS previously stored on the server 106 from the original product purchase package. If the count exceeds the predetermined drift threshold, then the routine returns affirmatively (step 1606). If not, then it returns negatively (step 1608).

Returning to the reader system flow as illustrated in Fig. 13, after the reader system 106 uploads the re-validation package to the license server 104, in a step 1306, the reader system 106 receives the re-validation result. If the re-validation was unsuccessful (step 1308), then the reader system displays an error message to the user and requests the customer to call customer service of the licensing authority (step 1310). In this situation, automatic re-validation has failed, and manual re-validation as in step 1310 is necessary. During the call, a customer service representative can determine whether the customer's license should be extended to cover the reader system 106 as it now stands. If automatic re-validation was successful (step 1308), then the reader system returns to step 514 (as indicated by the numeral "5" in the small circle in both Figs. 13 and 5) to store and process the new product server download package in the same manner as it processed the original product server download package received upon purchase.

It can be seen that a secure product distribution mechanism has been described which takes advantage of the benefits of an uncontrolled distribution network, while ensuring that authors and proprietors of digital products are paid an appropriate royalty for their efforts at creativity. In addition, the mechanism ensures that once a customer is licensed to use a

CLAIMS

1. A method for preparing a digital product for controlled distribution using a distribution network, comprising the steps of:

encrypting said product;

separating at least one encrypted fragment from said encrypted product;

transmitting said encrypted product less said at least one encrypted fragment onto said distribution network; and

withholding said at least one encrypted fragment from said distribution network.

2. A method according to claim 1, further comprising the step of transmitting said at least one encrypted fragment to a license server.

3. A method according to claim 2, further comprising the step of transmitting to said license server a decryption key that can be used to decrypt said product.

4. A method according to claim 1, wherein said encrypted product includes a header portion followed by a remainder portion,

and wherein said step of separating at least one encrypted fragment from said encrypted product comprises a step of separating from said encrypted product an encrypted fragment that includes at least part of said header portion.

8. A method according to claim 7, wherein less than all of said digital product is accessible to said server.

9. A method according to claim 7, wherein said product request information includes payment information,

further comprising a step of transmitting paid usage parameters to said user in response to said receipt of said product request information.

10. A method according to claim 7, wherein said product request information further includes a digest of a portion of a version of said digital product, said portion being non-co-extensive with said at least one fragment, further comprising the steps of:

storing a digest of said portion accessibly to said server prior to said step of receiving product request information; and

in response to receipt of said product request information, comparing said digest in said product request information with said digest stored accessibly to said server.

11. A method according to claim 7, wherein said product request information further includes a reader system signature of a particular reader system,

further comprising a step of encrypting at least one of said at least one fragment as stored accessibly to said server, in dependence upon said reader system signature, to form a further encrypted version of said at least one fragment as stored accessibly to said server,

14. A method according to claim 13, wherein said differences between said first and second reader system signatures satisfy said re-validation criteria, further comprising a step of encrypting at least one of said at least one fragment as stored accessibly to said server, in dependence upon said first reader system signature, to form a first further encrypted version of said at least one fragment,

the version transmitted to said user in said step of transmitting including said first further encrypted version of said at least one fragment;

said method further comprising the steps of:

encrypting said at least one of said at least one fragment as stored accessibly to said server, in dependence upon said second reader system signature, to form a second further encrypted version of said at least one fragment; and

transmitting said second further encrypted version of said at least one fragment to said user in response to said step of determining.

15. A method according to claim 13, wherein said differences between said first and second reader system signatures satisfy said re-validation criteria, further comprising the steps of:

storing a product decryption key accessibly to said server prior to said step of receiving product request information;

encrypting said product decryption key in dependence upon said first reader system signature, to form a first encrypted product decryption key; and

transmitting said first encrypted product decryption key to said user in response to receipt of said product request information,

step of using said digital product comprises a step of executing said software.

19. A method according to claim 16, wherein said first and second signatures do not satisfy said predetermined reader system drift criteria, further comprising the steps of:

manually communicating with a licensing authority for re-authorization; and

using said digital product at a time subsequent to said second time in response to said re-authorization.

20. A method according to claim 16, wherein said second reader system is said first reader system.

21. A method according to claim 16, wherein said step of using said digital product at said second time if and only if said first and second signatures satisfy predetermined reader system drift criteria, comprises the steps of:

said second reader system making a determination that said first and second signatures match; and

using said digital product in response to said determination.

22. A method according to claim 21, further comprising the steps of:

storing accessibly to said second reader system a first digest of at least a covered portion of said digital product; and

storing, at said first time and accessibly to said second reader system, at least one fragment of said digital product encrypted in dependence upon said

wherein said step of developing a first signature comprises the steps of:

developing a first component signature of each respective component in said first group as present in said first reader system at said first time; and

combining said first component signatures into a first combined signature.

26. A method according to claim 25, wherein said second group includes more than one component, and wherein said step of developing a second signature comprises the steps of:

developing a second component signature of each respective component in said second group as present in said second reader system at said second time; and

combining said second component signatures into a second combined signature,

wherein said first and second combined signatures constitute numerical combinations,

and wherein said step of using said digital product at said second time if and only if said first and second signatures satisfy predetermined reader system drift criteria, comprises the step of using said digital product at said second time if and only if said second combined signature differs from said first combined signature by no more than a predetermined maximum drift percentage.

27. A method according to claim 25, wherein said step of combining comprises a step of concatenating said first component signatures into said first combined signature,

wherein said second group includes more than one component,

inserting into said digital product a fingerprint that substantially uniquely identifies said second reader system, said fingerprint being recoverable from said digital product; and

using said digital product with said fingerprint inserted.

31. A method according to claim 16, wherein said step of using said digital product comprises the steps of:

inserting into said digital product a watermark that substantially uniquely identifies said second reader system, said watermark being recoverable from said digital product; and

using said digital product with said watermark inserted.

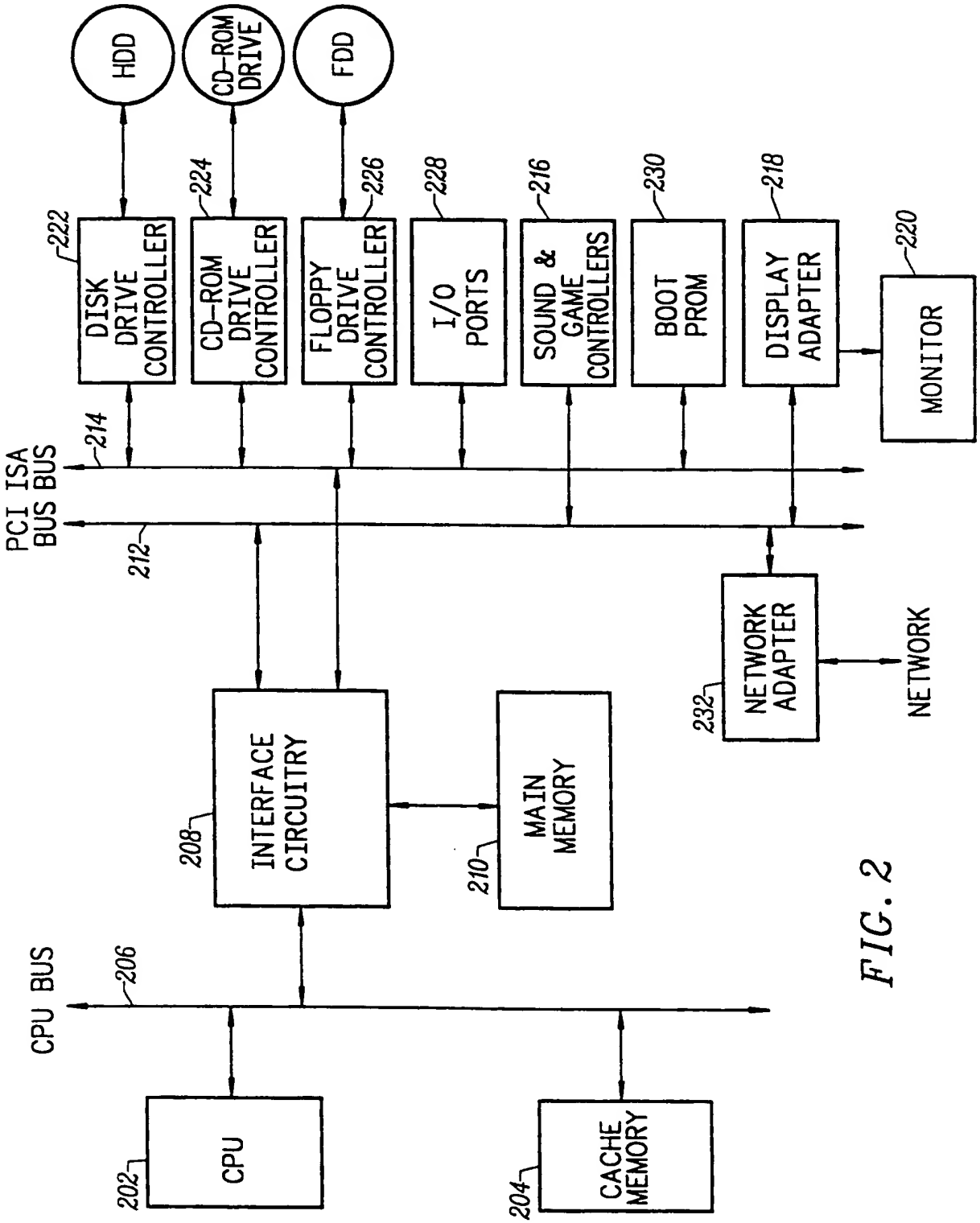

HDD
CD-ROM DRIVE
FDD
222
DISK DRIVE CONTROLLER
224
CD-ROM DRIVE CONTROLLER
226
FLOPPY DRIVE CONTROLLER
228
I/O PORTS
216
SOUND & GAME CONTROLLERS
230
BOOT PROM
218
DISPLAY ADAPTER
220
MONITOR
214
PCI ISA BUS BUS
212
NETWORK ADAPTER
232
NETWORK
208
INTERFACE CIRCUITRY
210
MAIN MEMORY
CPU BUS
206
202
CPU
204
CACHE MEMORY

FIG. 2

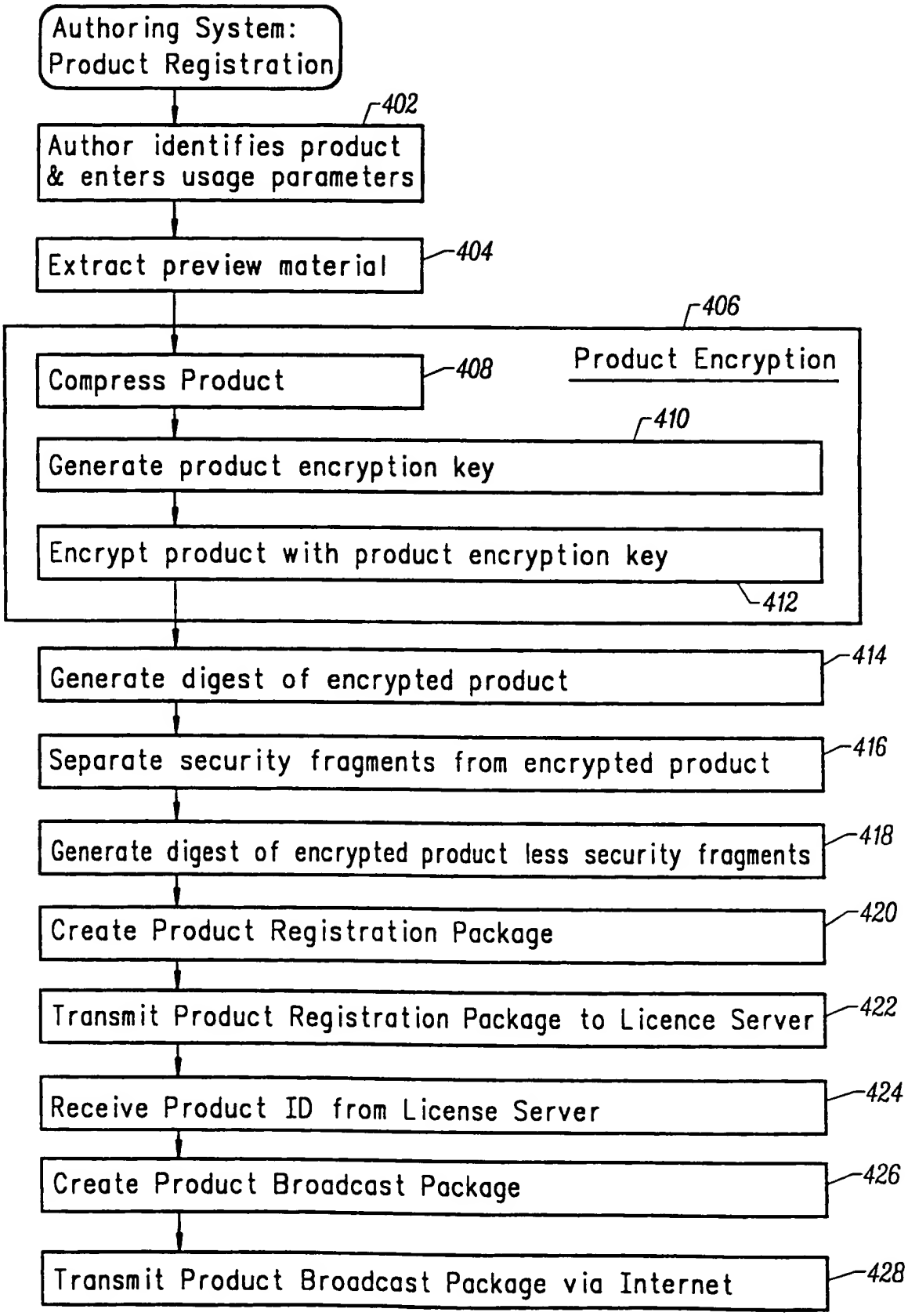

Authoring System: Product Registration
402
Author identifies product & enters usage parameters
Extract preview material
404
406
Product Encryption
Compress Product
408
410
Generate product encryption key
Encrypt product with product encryption key
412
Generate digest of encrypted product
414
Separate security fragments from encrypted product
416
Generate digest of encrypted product less security fragments
418
Create Product Registration Package
420
Transmit Product Registration Package to Licence Server
422
Receive Product ID from License Server
424
Create Product Broadcast Package
426
Transmit Product Broadcast Package via Internet
428

FIG. 4

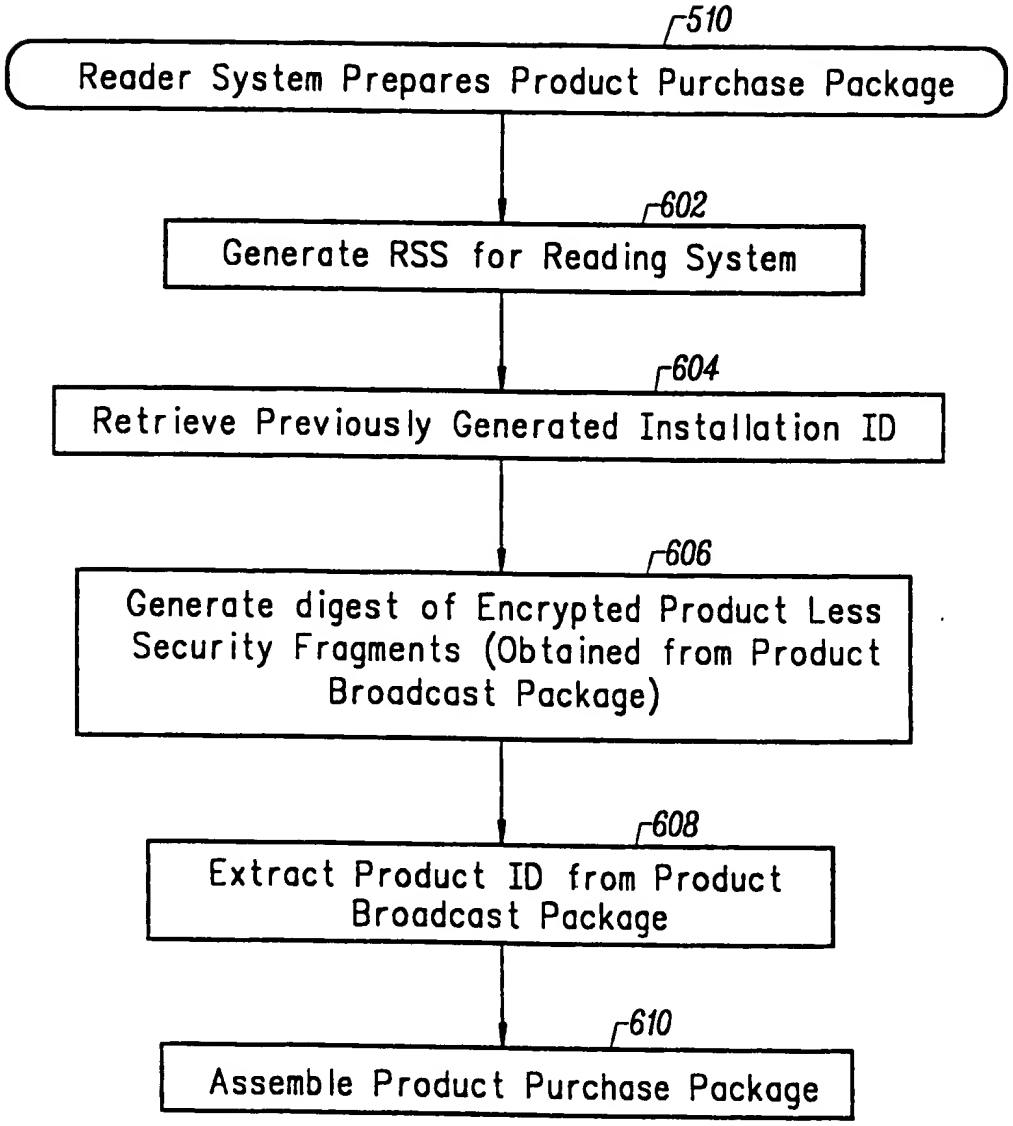

510
Reader System Prepares Product Purchase Package
602
Generate RSS for Reading System
604
Retrieve Previously Generated Installation ID
606
Generate digest of Encrypted Product Less Security Fragments (Obtained from Product Broadcast Package)
608
Extract Product ID from Product Broadcast Package
610
Assemble Product Purchase Package

*FIG. 6*

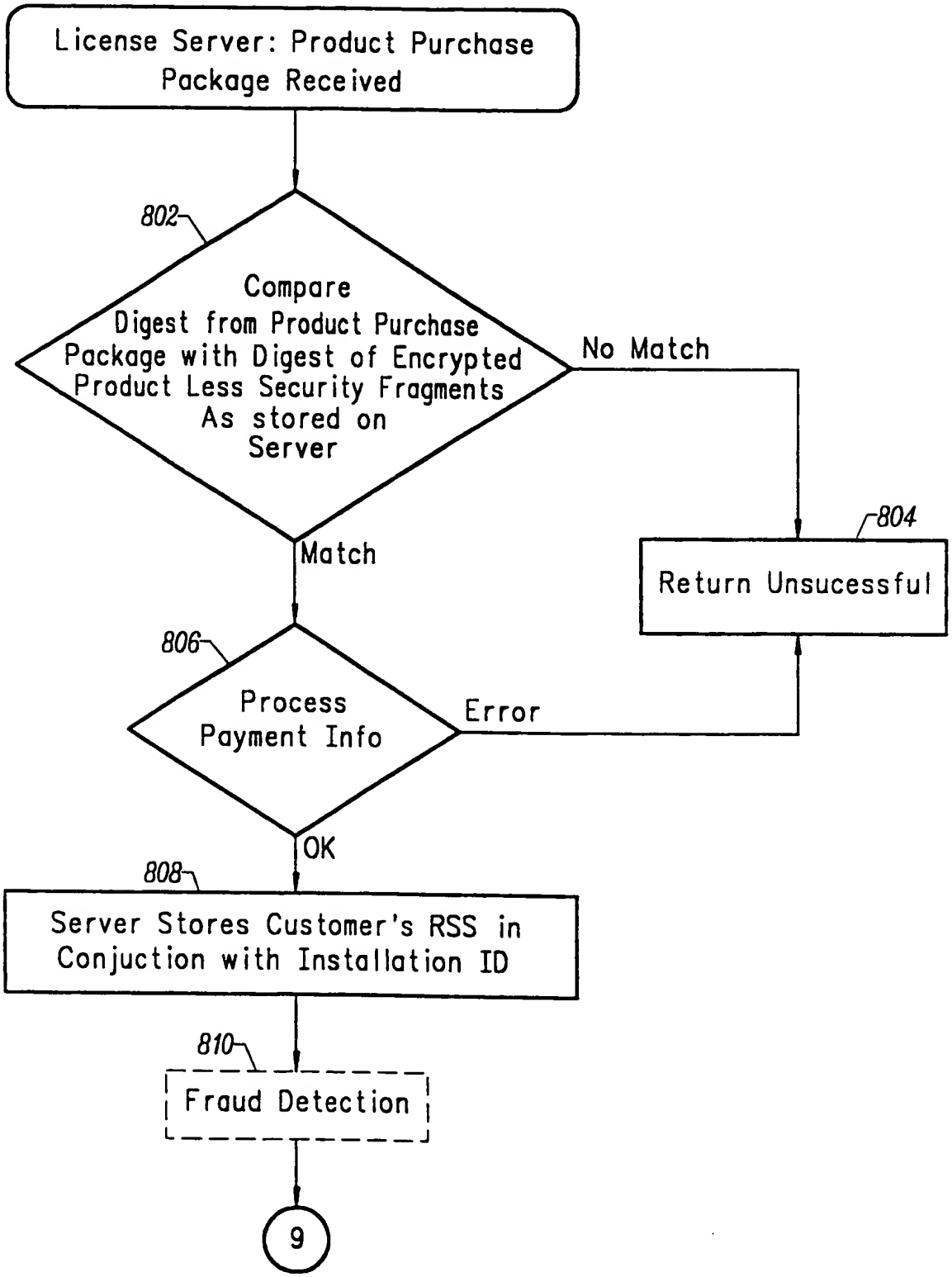

License Server: Product Purchase Package Received
802
Compare Digest from Product Purchase Package with Digest of Encrypted Product Less Security Fragments As stored on Server
No Match
804
Return Unsucessful
Match
806
Process Payment Info
Error
OK
808
Server Stores Customer's RSS in Conjuction with Installation ID
810
Fraud Detection
9

FIG. 8